CLAIMS

1. A method of transmitting data securely over a computer network, comprising the steps of:

(1) establishing a communication path between a first computer and a second computer;

(2) encrypting and transmitting data records between the first computer and the second computer using an unreliable communication protocol, wherein each data record is encrypted without reference to a previously transmitted data record;

(3) in the second computer, receiving and decrypting the data records transmitted in step (2) without reference to a previously received data record; and

(4) in the second computer, transmitting session information for encrypting and decrypting the data records to a third computer.

2. The method of claim 1, further comprising the step of, prior to step (1), establishing a reliable communication path between the first computer and the second computer and exchanging security credentials over the reliable communication path.

3. The method of claim 2, wherein the step of exchanging security credentials comprises the step of exchanging an encryption key that is used to encrypt the data records in step (2).

4. The method of claim 2, wherein the session information includes at least a portion of the security credentials.

5. The method of claim 1, wherein step (2) comprises the step of incorporating a nonce in each data record that is used by the second computer in combination with a previously shared encryption key to decrypt each of the data records in step (3).

6. The method of claim 5, wherein the nonce comprises a random number.

7. The method of claim 5, further comprising the step of, in the second computer, verifying that the nonce has not previously been received in a previously transmitted data record.

8. The method of claim 1,

wherein step (2) comprises the step of embedding an indicator in each of the data records indicating that the data records are encrypted according to an encryption scheme that encrypts records without regard to any previously transmitted data records, and

wherein step (3) comprises the step of determining whether the indicator is present in

each record and, in response to determining that the indicator is not present, processing each such record differently than if the indicator is set.

9. The method of claim 1, wherein step (1) is performed using the Transmission Control Protocol, and wherein step (2) is performed using the User Datagram Protocol.

10. The method of claim 1, wherein step (2) is performed by a proxy server that encrypts data records received from another server.

11. The method of claim 1, wherein the third computer

establishes a communication path with the first computer; and

encrypts and transmits data records to the first computer using an unreliable communication protocol, wherein each data record is encrypted without reference to a previously transmitted data record and by employing the session information.

12. The method of claim 1, wherein a fourth computer

retrieves the session information from the third computer'

establishes a communication path with the first computer; and

encrypts and transmits data records to the first computer using an unreliable communication protocol, wherein each data record is encrypted without reference to a previously transmitted data record and by employing the session information.

13. The method of claim 1, wherein the session information is SSL or TLS session information.

14. The method of claim 1, wherein the session information includes a SSL or TLS session identifier.

15. The method of claim 1, wherein the session information includes an encryption key that is used to encrypt data records in step (2).

16. The method of claim 1, wherein the session information is stored by the third computer in a cache memory using a hash function.

17. The method of claim 16, wherein the hash function is the BUZhash function.

18. The method of claim 1, wherein the second computer transmits the session information to the third computer using multicast communication.

19. The method of claim 18, wherein the multicast communication is negative

acknowledgement multicast communication.

20. A method of securely transmitting a plurality of data records between a client computer and a proxy server using an unreliable communication protocol, comprising the steps of:

(1) establishing a reliable connection between the client computer and the proxy server;

(2) exchanging encryption credentials between the client computer and the proxy server over the reliable connection;

(3) generating a nonce for each of a plurality of data records, wherein each nonce comprises an initialization vector necessary to decrypt a corresponding one of the plurality of data records;

(4) using the nonce to encrypt each of the plurality of data records and appending the nonce to each of the plurality of data records;

(5) transmitting the plurality of data records encrypted in step (4) from the client computer to the proxy server using an unreliable communication protocol;

(7) in the proxy server, decrypting each of the plurality of encrypted data records using a corresponding nonce extracted from each data record and a previously shared encryption key; and

(8) in the proxy server, transmitting session information including the previously shared encryption key for use in decrypting the plurality of data records to another server.

21. The method of claim 20, wherein step (6) comprises the step of checking to determine whether each data record received from the client computer is formatted according to a secure unreliable transmission format and, if a particular record is not formatted according to a secure unreliable transmission format, bypassing the decryption using the corresponding nonce.

22. The method of claim 20, wherein step (3) comprises the step of generating a random number as each nonce.

23. The method of claim 20, wherein step (3) comprises the step of generating an unique number as each nonce.

24. The method of claim 20, wherein step (1) is performed using Transmission Control Protocol, and wherein step (5) is performed using User Datagram Protocol.

25. The method of claim 20, wherein step (6) is performed using an encryption key

previously shared using a reliable communication protocol.

26. The method of claim 25, wherein the reliable communication protocol is Transmission Control Protocol.

27. The method of claim 20, wherein the another server is a second proxy server.

28. The method of claim 27, further including, in the second proxy server, decrypting encrypted data records from the client computer using a corresponding nonce extracted from each data record and the session information transmitted from the first proxy server.

29. The method of claim 20, wherein the another proxy server is a cache memory server.

30. The method of claim 29, further including, in a second proxy server,

obtaining the session information from the cache memory server, and

decrypting encrypted data records from the client computer using a corresponding nonce extracted from each data record and the session information.

31. The method of claim 20, wherein the session information is SSL OR TLS session information.

32. The method of claim 20, wherein the session information includes a SSL OR TLS session identifier.

33. The method of claim 20, wherein the session information includes authentication information for a user of the client computer.

34. The method of claim 20, wherein the session information is stored by the other server in a cache memory using a hash function.

35. The method of claim 34, wherein the hash function is the BUZhash function.

36. The method of claim 20, wherein the another proxy server transmits the session information to the other using multicast communication.

37. The method of claim 36, wherein the multicast communication is negative acknowledgement multicast communication.

38. A system for securely transmitting data using an unreliable protocol, comprising:

a first computer comprising a communication protocol client function operable in conjunction with an application program to transmit data records securely using an unreliable protocol; and

a second computer coupled to the first computer and comprising a communication protocol server function operable in conjunction with the communication protocol client function to receive data records securely using the unreliable communication protocol,

wherein the communication protocol client function encrypts each data record using a nonce and an encryption key and appends the respective nonce to each of the encrypted data records; and

wherein the communication protocol server function decrypts each of the data records using the respectively appended nonce and the encryption key; and

a third computer coupled to the second computer and having a cache memory for storing at least the encryption key.

39. The system of claim 38, wherein the communication protocol client function exchanges encryption credentials with the communication protocol server function using a reliable communication protocol.

40. The system of claim 39, wherein the unreliable communication protocol comprises the User Datagram Protocol, and wherein the reliable communication protocol comprises the Transmission Control Protocol.

41. The system of claim 38, wherein the communication protocol client function and the communication protocol server function are compatible with the SOCKS communication protocol.

42. The system of claim 38, wherein the communication protocol client function and the communication protocol server function are compatible with the SSL/TLS communication protocol.

43. The system of claim 38, wherein the second computer comprises a proxy server that forwards decrypted records received from the first computer to a server computer.

44. The system of claim 38, wherein the second computer comprises a record detector that determines whether an indicator has been set in each data record received from the first computer and, if the indicator has not been set, bypassing decryption in the server computer.

45. The system of claim 38, wherein the third computer is a proxy server that can receive encrypted records from the first computer;

can decrypt records the received records using at least the encryption key stored in the cache memory; and

can forward the decrypted records received from the first computer to a server computer.

46. The system of claim 38, wherein the third computer is a memory cache server, and further including a fourth computer that can

obtain the at least the encryption key stored in the cache memory of the third compter;

receive encrypted records from the first computer;

decrypt records the received records using at least the encryption key stored in the cache memory; and

forward the decrypted records received from the first computer to a server computer.